REMARKS

Claims 21-28 are pending, with claim 21 being independent. Reconsideration and allowance of the above-referenced application are respectfully requested.

Rejections under 35 U.S.C. 103

Claims 21-28 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Kouznetsov (U.S. Patent 6,973,577), and further in view of Gryaznov (U.S. Patent 7,065,790). This contention is respectfully traversed.

Claim 21 recites, in part, "one or more machines coupled with the network, each machine comprising a communication interface and a memory including an execution area configured to perform operations comprising examining a set of instructions embodying an invoked application to identify the invoked application, obtaining application-specific intrusion criteria, and monitoring network communications for the invoked application, after the examining and the obtaining, using the application-specific intrusion criteria to detect an intrusion." In addressing this claim language, the Office states[1]:

> configured to perform operations comprising examining a set of instructions embodying an invoked application (col.2, lines 47-48 and col.4, lines 12-14 and 28-47) *[to identify the invoked application]*, obtaining application-specific intrusion criteria (col.2, lines 51-58 and col.5, lines 9-12 and col.7, lines 1-2), and monitoring network communications for the invoked application, after the examining and the obtaining, using the application-specific intrusion criteria to detect an intrusion. (col.2, lines 32-40 and col.4, lines 15-36)

[1] *See* 6-20-2008 Office Action at page 3.

However, Kouznetsov fails to teach or suggest the claimed subject matter, either alone or in combination with Gryaznov. Kouznetsov teaches a system and a method for dynamically detecting computer viruses through associative behavioral analysis of runtime state.[2] But Kouznetsov does not in any way teach obtaining application-specific intrusion criteria.

In rejecting this claimed feature, the Office cites to various portions of Kouznetsov that have nothing to do with application-specific intrusion criteria[3]:

> The sequence of the execution of the monitored events is tracked for each of the applications. Each occurrence of a specific event sequence characteristic of computer virus behavior and the application that performed the specific event sequence, are identified. A histogram describing the specific event sequence occurrence for each of the applications is created. Repetitions of the histogram associated with at least one object are identified.
>
> The process identifier (ID) 71 and application name 72 fields respectively store the process number and name of the application 33, 34, 35 (shown in FIG. 2) to which the recorded monitored event is associated.
>
> ... records for the monitored events 70 are retrieved for each of the applications 33, 34, 35 (block 151).

Nothing in these portions of Kouznetsov, nor any other portion of Kouznetsov, describes obtaining intrusion criteria that are application-specific. While it is true that Kouznetsov tracks the sequence of the execution of monitored events for each of multiple applications, Kouznetsov does not describe obtaining intrusion criteria that are specific to these various applications. In fact, Kouznetsov arguably teaches the opposite of this.

---

[2] *See* Kouznetsov at Abstract.
[3] *See* Kouznetsov at col. 2, lines 51-58 and col. 5, lines 9-12 and col. 7, lines 1-2.

Kouznetsov teaches that the program state of the executing applications is monitored by a monitor/analyzer 19 that monitors all applications equally using apparently common criteria[4]:

> The program state of the executing applications 33, 34, 35 is monitored by the monitor/analyzer 19 that generates histograms based on occurrences of monitored events. The monitor/analyzer 19 functions as a logical "shim" interposed between the operating system 32 and each of the applications 33, 34, 35. Each system call is intercepted by the monitor/analyzer 19 which compares the requested system call to a list of monitored events. If the system call matches one of the monitored events, the monitor 19 determines whether the application is performing a sequence of "suspicious" actions characteristic of computer viruses. If so, histograms of the event occurrences are generated and stored in a database 37 maintained in the storage device 36.

Thus, Kouznetsov does not in any way teach the claimed, "obtaining application-specific intrusion criteria". On this basis alone, all of the current rejections should be withdrawn.

In addition, neither Kouznetsov nor Gryaznov, either alone or in combination, teaches or suggests, "examining a set of instructions embodying an invoked application to identify the invoked application", as claimed. The cited portions of Kouznetsov teach the following[5]:

> Each action is performed by one or more applications executing within a defined computing environment.
>
> A plurality of applications 33, 34, 35 are loaded into the RAM from storage device 36 and executed by the CPU.
>
> Alternatively, the histograms could be stored in a centralized database for analysis of distributed runtime state, such as described in the related, commonly-assigned U.S. Patent application, entitled "System And Method For Dynamically Detecting Computer Viruses Through Behavioral Analysis Of Distributed Runtime State Using An Analysis

[4] *See* Kouznetsov at col. 4, lines 15-27.
[5] *See* Kouznetsov at col. 2, lines 47-48 and col. 4, lines 12-14 and 28-47.

Tree," filed May 26, 2000, pending, the disclosure of which is incorporated herein by reference. The histograms are analyzed to identify repetitions of suspect behavior.

FIG. 3 is a block diagram showing the functional software modules 50 of the monitor/analyzer 19 of FIG. 1. Each module is a computer program written as source code in a conventional programming language, such as the C or C++ programming languages, and is presented for execution by the CPU as object or byte code, as is known in the art. The various implementations of the source code and object and byte codes can be held on a computer-readable storage medium or embodied on a transmission medium in a carrier wave.

Interpreting the present claim language, "examining a set of instructions embodying an invoked application" as reading on the traditional loading and executing of program code is inconsistent with the present specification and is thus an improper claim construction under the law and the MPEP.[6]

Moreover, the Office's use of Gryaznov in combination with Kouznetsov to address the full claim feature, "examining a set of instructions embodying an invoked application to identify the invoked application", is without merit. The Office has split this claim feature into two parts: (1) "examining a set of instructions embodying an invoked application", and (2) "to identify the invoked application"; and the Office then attemps to combine Gryaznov with Kouznetsov to arrive at the full claim feature. Gryaznov describes a method, system, and computer program product that provides multiple names of a given malware in a quick and automated fashion.[7] As noted above, and in sharp contrast, Kouznetsov teaches a system and a method for dynamically detecting computer viruses through associative behavioral analysis of runtime state. The Office

[6] *See e.g.*, MPEP § 2111.01.
[7] *See* Gryaznov at Abstract.

has provided no articulated reasoning with some rational underpinning to combine these two references to support the legal conclusion of obviousness, as is required by law.

Rather, the apparent reason the Office is combining the references is to piece together the elements of the claimed subject matter from the prior art. This is blatant hindsight reconstruction and is not allowed under the governing laws. The Office is respectfully reminded that, "in formulating a rejection under 35 U.S.C. § 103(a) based upon a combination of prior art elements, it remains necessary to identify the reason why a person of ordinary skill in the art would have combined the prior art elements in the manner claimed."[8] In light of this, and the arguments presented above, it should now be clear that the Office has failed to meet its burden of providing a prima facie showing of obviousness in the present application.

Thus, independent claim 21 should be in condition for allowance. Dependent claims 22-28 should be patentable based on the above arguments and the additional recitations they contain. No fees are believed due with this response. Nonetheless, please apply any necessary charges or credits to deposit account 06-1050.

Respectfully submitted,

Date: Sept. 22, 2008

William E. Hunter
Reg. No. 47,671

Fish & Richardson P.C.
PTO Customer No. 20985
Telephone: (858) 678-5070
Facsimile: (877) 769-7945

10847125.doc

[8] *See* Memorandum dated May 3, 2007, to Technology Center Directors from Margaret A. Focarino, Deputy Commissioner for Patent Operations, re Supreme Court decision on KSR Int'l. Co., v. Teleflex, Inc. (emphasis added).